CLAIMS

What is claimed is:

1. A register renaming system for out-of-order execution of a set of reduced instruction set computer instructions having addressable source and destination register fields, adapted for use in a computer having an instruction execution unit with a register file accessed by read address ports and for storing instruction operands, the system comprising:

(a) data dependance check means for determining data dependencies between the instructions;

(b) tag assignment means for generating one of more tags to specify the location of operands, based on said data dependencies determined by said data dependance check means; and

(c) register file port means for selecting said tags generated by said tag assignment means and passing said tags onto the read address ports of the register file for storing execution results.

2. The system of claim 1, wherein said data dependance check means determines said data dependencies by comparing the addresses of the source register field of each instruction to the addresses of the destination register fields.

3. The system of claim 1, further comprising temporary storage means for temporarily storing out-of-order execution results, wherein said out-of-order execution results are passed to the register files in order after execution of the set of instructions is completed.

4. A register renaming method for performing out-of-order execution of a set of reduced instruction set computer instructions having addressable source and destination register fields, adapted for use in a computer having an instruction execution unit with a register file accessed by read address ports and for storing instruction operands, the method comprising the steps of:

(1) determining data dependencies between the instructions;

(2) performing at least one of in-order and out-of-order issuing of two or more of the instructions in the instruction execution unit;

(3) storing, temporarily, any out-of-order results in temporary storage means;

(4) generating one or more tags to specify the location of said out-of-order results based on said data dependencies;

(5) selecting appropriate ones of said tags corresponding to the issued instruction; and

(6) passing said selected tags onto the read address ports of the register file for one of:

(i) accessing said out-of-order results; and

(ii) storing execution results.

5. The method of claim 4, wherein said determining step further comprises the step of comparing the addresses of the source register field of each instruction to the addresses of the destination register fields.

6. The method of claim 4, further comprising the step of storing in-order results in the register file and the temporary storage means.

7. The method of claim 4, further comprising the step of passing said out-of-order results to the register files in-order after execution of the set of instructions is completed.

8. A method for issuing instructions in a superscalar reduced instruction set computer having an instruction issuer system capable of issuing a plurality of instructions in a single cycle, the system having more than one register file sets, the method comprising the steps of:

prioritizing instructions to be issued according to the number of different register file sets furnishing operands to the instruction; and

issuing instructions according to said prioritizing step.

9. The method according to claim 8, wherein said prioritizing step assigns a higher priority to those instructions having operands furnished from the greater number of register file sets.

10. An instruction issuer system in a superscalar reduced instruction set computer, the system capable of issuing a plurality of instructions in a single cycle, the system having more than one set of register files, the system comprising:

first means for prioritizing instructions to be issued according to the number of different register files furnishing operands to the instruction; and

second means, responsive to said first means, for issuing instructions according to said priority.

11. The system according to claim 10, wherein said first means further assigns a higher priority to those instructions having operands furnished from the greater number of register files.

add B1